

FIG. 1A

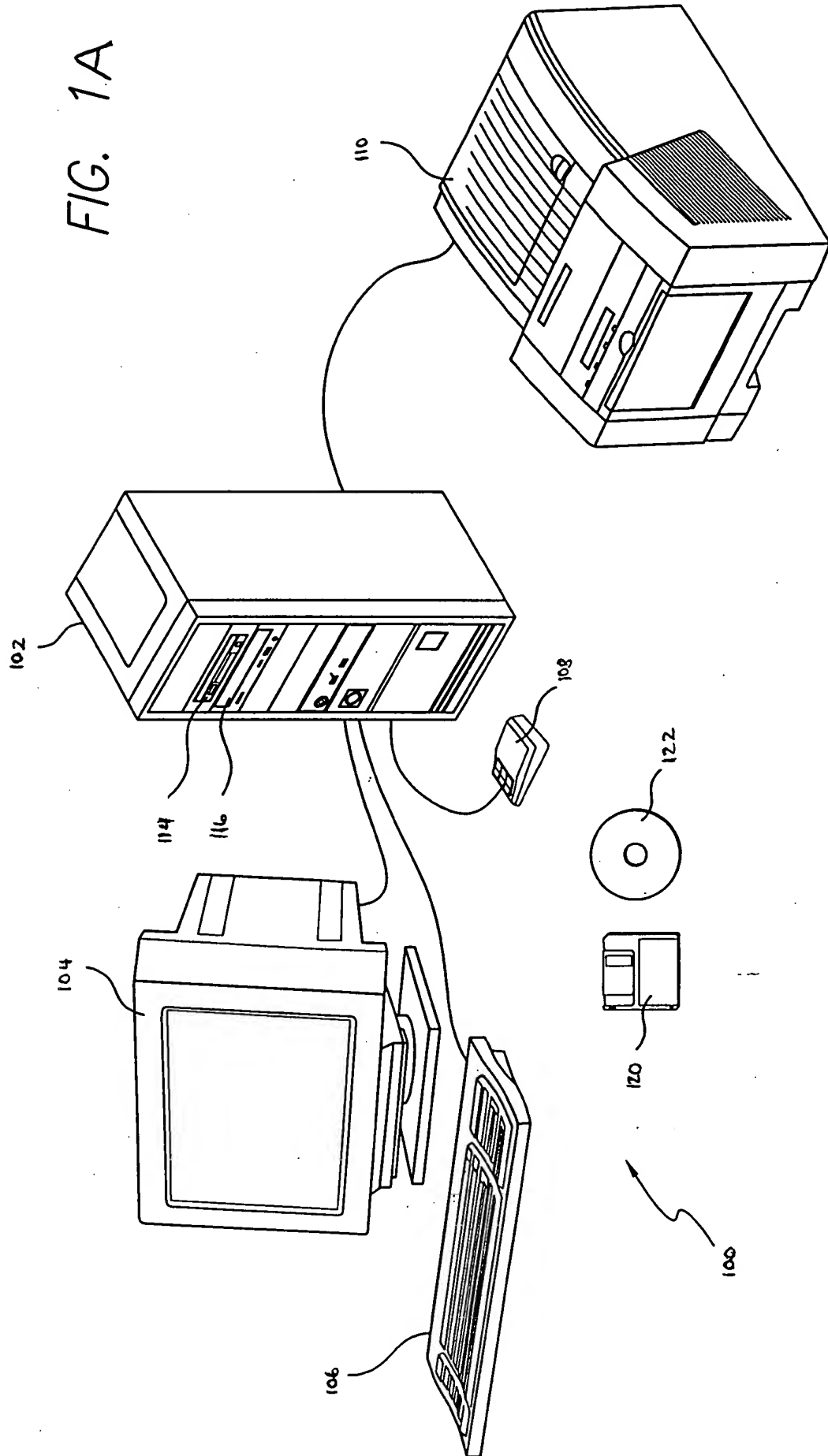
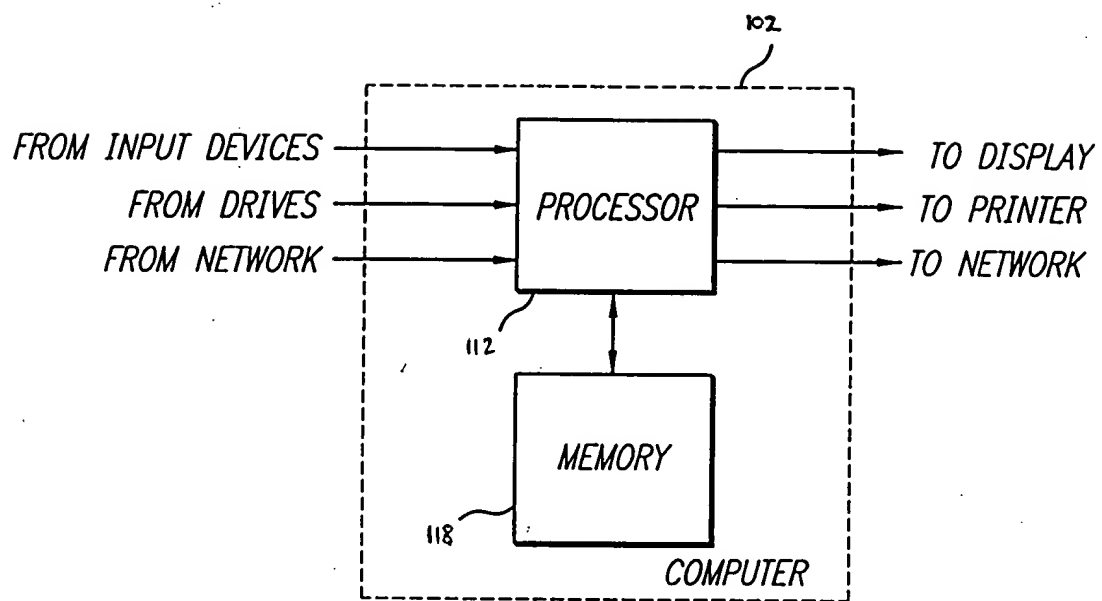


FIG. 1B



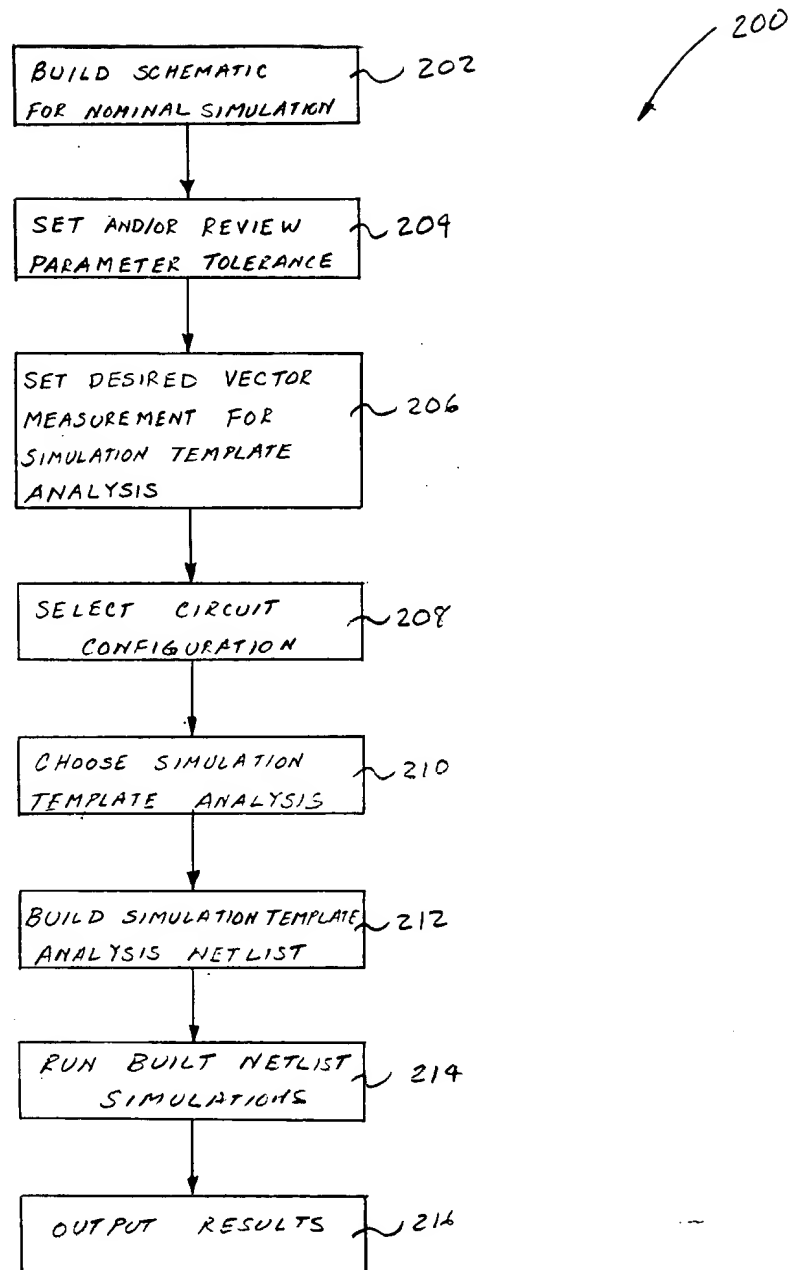


FIGURE 2

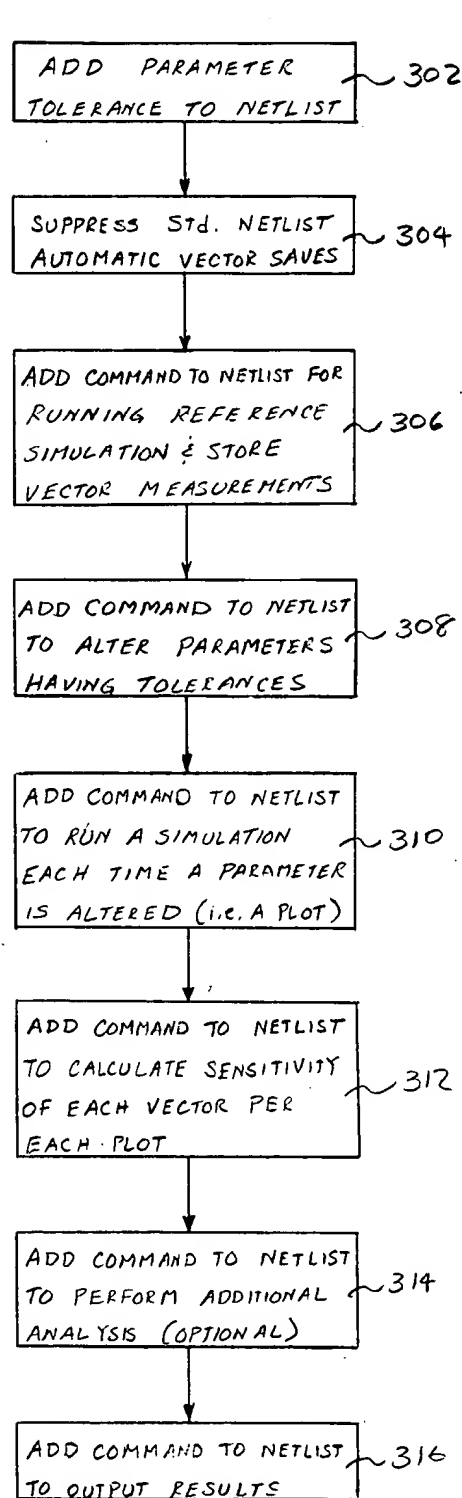
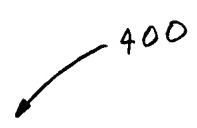


FIGURE 3

**SENS, sensitivity analysis Simulation Template With Comments:**

FIGURE 4-1



```

*Instruct the netlist builder to show tolerances } 402
#tolerance

*Suppress automatic vector saves } 404
#nosave

*Suppress IsSpice4 printout } 406
#noprint

*Save vectors needed for measurements } 408
#vector

*Set the output file pointer to the beginning to remove } 410
* the input net list
set rewind

*Set the noecho environment for print formatting } 412
set noecho

*Run the specified simulation and save the results } 414
#simulation
set printmode = save
#mprint

*Rename the simulation plot } 416
nameplot ref

*Set the print format } 418
SET COLWIDTH=22
SET SPICEDIGITS=5

*Tell the user where we are
printstatus -t #####_sensitivity_for_each_parameter_##### } 420

*Loop through all of the parameters
nextparam null
while param <> null
    *Alter each parameter
    alterparam tolerance (param) /3
    *Simulate
    #simulation
    Save the parameter reference in the new plot
    paramvec = param
    *Tell the user where we are
    printstatus -p paramvec
    *Save the data
    #mprint
    *Loop through the vectors
    nv = nextvector(null)
    while nv <> null
        *Save the sensitivities of scalar data
        if length(nv) = 1
            nv = nv - ref.nv

```

FIGURE 4-2

```

        end; end if
        *Get the next vector
        nv = nextvector(nv)
    end ; end vector loop
    *Restore the parameter
    unalterparam
    *Get the next parameter
    nextparam
end; end parameter loop
*Set print mode for printing output data
set printmode = print
unset noecho
*Loop through the plots
pl = nextplot(null)
while pl <> null
    if sameplot(ref.default) = 0
        *Loop through the vectors
        nv = nextvector(null)
        while nv <> null
            if length(nv) = 1
                *Save the sensitivities
                if ref.nv <> 0
                    nv = (300*Nv)/REF.NV
                else
                    nv = 3*Nv
                end ; end if
            end ; end if
            nv = nextvector(nv)
        end ; end vector loop
    end; end if
    *Get next plot
    pl = nextplot(pl)
end ; end plot loop
*Tell the user what's going on
printstatus -t #####sorting_sensitivity_for_each_parameter_##
printstatus -t
*Make ref the current plot
setplot ref
*Sort by descending value
sort -vd
*Loop through the plots
pl = nextplot(null)
while pl <> null
    if sameplot(ref.default) = 0
        *Print Headers
        SETPARAM PARAMVEC
        printstatus -p paramvec
        ECHO
        ECHO -u "*****SENSITIVITY DATA*****"
        ECHO
        ECHO -un "PARAMETER NAME: "
        PRINTNAME PARAMVEC
        ECHO
        ECHO -un " NOMINAL VALUE: "
    end ; end if
end ; end plot loop

```

424

426

428

FIGURE 4-3

```

PRINTVAL PARAMVEC
ECHO
ECHO
PRINTTEXT -u VECTOR SENSITIVITY%
ECHO
ECHO
*Sort by descending data value
sort -vd
*Loop through the vectors and print data
nv = nextvector(null)
while nv <> null
    if length(nv) = 1
        if ref.nv <> 0
            PRINTNAME NV
            PRINTVAL NV
        else
            PRINTNAME NV
            PRINTVAL NV
            ECHO -n *
        end ; end if
    end ; end if
    *Get next vector
    nv = nextvector(nv)
end; end vector loop
end ; end if
*Get next plot
pl = nextplot(pl)
end; end plot loop
ECHO
ECHO
*Print data in output file for SpiceNet to read
setplot ref
echo ##### SENSITIVITY analysis Results #####
#mprint

```

430

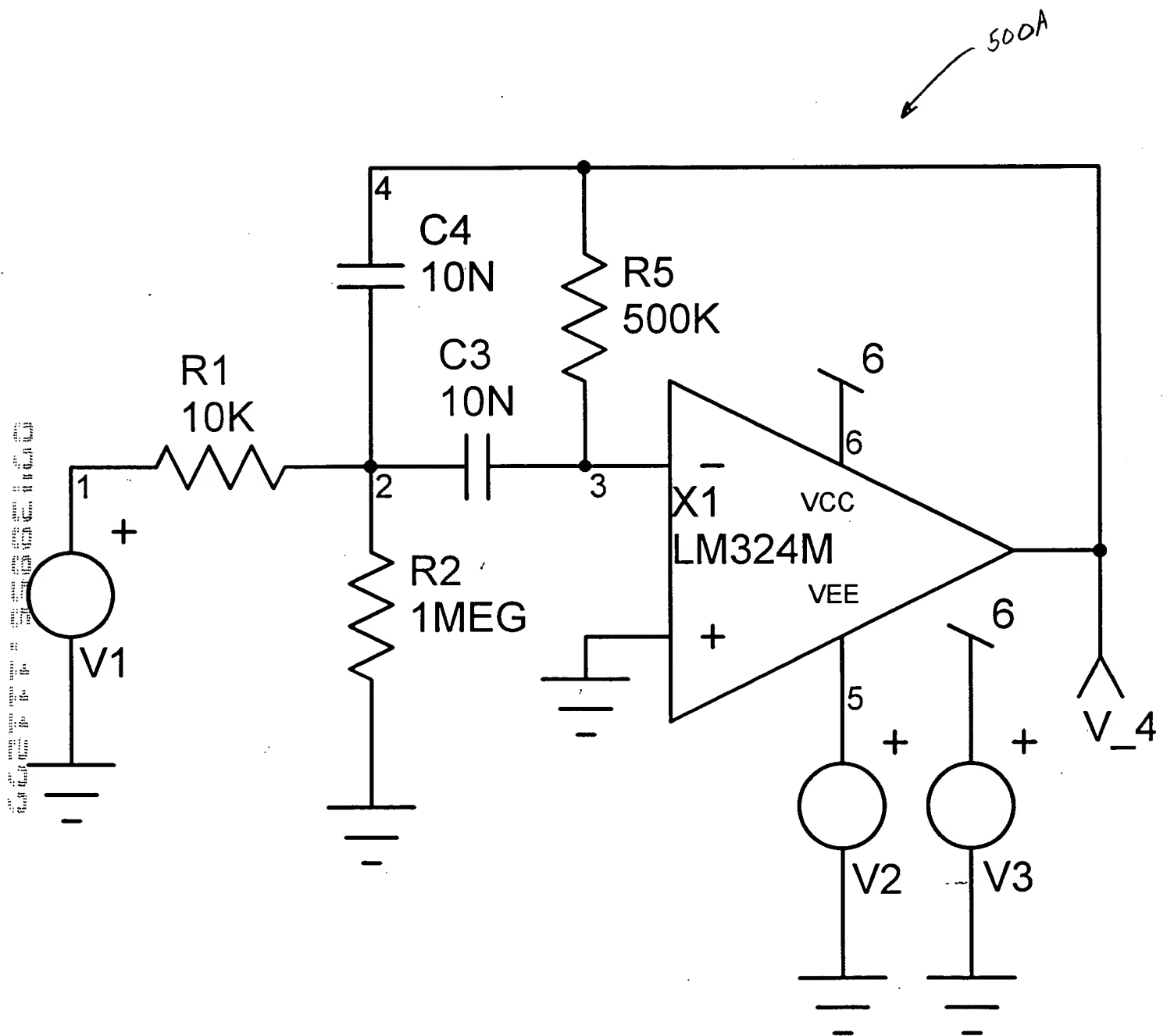


FIGURE 5A



```

C:\spice8d\Circuits\Bandpass.cir Setup1
*#save V(2) V(3) @C3[i] @C3[p] V(1) @R1[i] @R1[p] V(4)
*#save @C4[i] @C4[p] @R2[i] @R2[p] @R5[i] @R5[p] V(6) V(5)
*#save @V1[i] @V1[p] @V2[i] @V2[p] @V3[i] @V3[p]
*#alias v_4 v(4)
*#view tran v_4
.TRAN .05ms 20ms
.PRINT AC VDB(4)
.OPTIONS vscale=4
.PRINT TRAN V_4
C3 2 3 10N
R1 1 2 10K
C4 2 4 10N
R2 2 0 1MEG
R5 3 4 500K
X1 0 3 6 5 4 LM324M { }
.SUBCKT LM324M 1 2 3 4 5
*
C1 11 12 3.000E-12
C2 6 7 6.000E-12
CEE 10 99 315.8E-15
DC 5 53 DX
DE 54 5 DX
DLP 90 91 DX
DLN 92 90 DX
DP 4 3 DX
EGND 99 0 POLY(2) 3 0 4 0 0 .5 .5
FB 7 99 POLY(5) VB VC VE VLP VLN 0 53.05E6
+ -50E6 50E6 50E6 -50E6
GA 6 0 11 12 37.70E-6
GCM 0 6 10 99 11.92E-9
IEE 3 10 DC 2.476E-6
HLIM 90 0 VLIM 1K
Q1 11 2 13 QX
Q2 12 1 14 QX
R2 6 9 100.0E3
RC1 4 11 26.53E3
RC2 4 12 26.53E3
RE1 13 10 4.820E3
RE2 14 10 4.820E3
REE 10 99 80.78E6
RO1 8 5 50
RO2 7 99 50
RP 3 4 34.71E3
VB 9 0 DC 0
VC 3 53 DC 2
VE 54 4 DC 5.000E-3
VLIM 7 8 DC 0
VLP 91 0 DC 40
VLN 0 92 DC 40
.MODEL DX D(IS=800.0E-18)
.MODEL QX PNP(IS=800.0E-18 BF=31.58)
.ENDS
V1 1 0 AC=1 PULSE 0 -1 1MS
V2 5 0 DC=-5
V3 6 0 DC=5
.END

```

502

500B

506

508

FIGURE 5B

```

C:\spice8d\Circuits\Bandpass.Cir Setup1
.OPTIONS vscale=4
.control
alias v_4 v(4)
view tran v_4
save v(4) — 608

```

FIGURE 6-1

```

set rewind — 610
set noecho — 612
TRAN .05ms 20ms

```

```

set printmode = save
echo TRAN Analysis Measurements
echo
echo Test 1 Mean
homeCursors
print Mean(V(4))

```

614

```
nameplot ref — 616
```

```

SET COLWIDTH=22
SET SPICEDIGITS=5 ] 618

```

```

printstatus -t #####_sensitivity_for_each_parameter_##### — 620
nextparam null

```

```

while param <> null
  alterparam tolerance(param)/3
  TRAN .05ms 20ms

```

```

  paramvec = param
  printstatus -p paramvec
  echo TRAN Analysis Measurements

```

```

echo
echo Test 1 Mean
homeCursors
print Mean(V(4))

```

622

```

  nv = nextvector(null)
  while nv <> null
    if length(nv) = 1
      nv = nv - ref.nv
    end
    nv = nextvector(nv)
  end
  unalterparam
  nextparam

```

```

end
set printmode = print
unset noecho

```

```

pl = nextplot(null)
while pl <> null
  if sameplot(ref.default) = 0

```

```
    nv = nextvector(null)
```

```
    while nv <> null
```

```
      if length(nv) = 1
```

```
        if ref.nv <> 0
```

```
          nv = (300*Nv)/REF.NV
```

```
        else
```

```
          nv = 3*Nv
```

```
        end
```

```
      end
```

```
      nv = nextvector(nv)
```

```
    end
```

```
    pl = nextplot(pl)
```

```
end
```

624

```

printstatus -t #####_sorting_sensitivity_for_each_parameter_#####
printstatus -t

```

626

```
setplot ref
```

```
sort -vd
```

```
pl = nextplot(null)
while pl <> null
  if sameplot(ref.default) = 0
```

```
    SETPARAM PARAMVEC
```

```
    printstatus -p paramvec
```

```
    ECHO
    ECHO -u "*****SENSITIVITY DATA*****"
    ECHO
    ECHO -un "PARAMETER NAME: "
    PRINTNAME PARAMVEC
    ECHO
    ECHO -un " NOMINAL VALUE: "
    PRINTVAL PARAMVEC
    ECHO
    ECHO
    PRINTTEXT -u VECTOR SENSITIVITY%
    ECHO
    ECHO
```

```
    sort -vd
    nv = nextvector(null)
```

```
    while nv <> null
      if length(nv) = 1
        if ref.nv <> 0
          PRINTNAME NV
          PRINTVAL NV
        else
          PRINTNAME NV
          PRINTVAL NV
          ECHO -n *
        end
      end
      ECHO
      end
      nv = nextvector(nv)
    end
```

```
  end
  pl = nextplot(pl)
end
```

```
ECHO
ECHO
```

```
setplot ref
echo ##### SENSITIVITY analysis Results #####
echo TRAN Analysis Measurements
echo
echo Test 1 Mean
homeCursors
print Mean(V(4))
```

```
.endc
```

```
C3 2 3 10N TOL=5%
R1 1 2 10K TOL=2%
C4 2 4 10N TOL=5%
R2 2 0 1MEG TOL=2%
R5 3 4 500K TOL=2%
X1 0 3 6 5 4 LM324M { }
.SUBCKT LM324M 1 2 3 4 5
*
```

```
C1 11 12 3.000E-12
C2 6 7 6.000E-12
CEE 10 99 315.8E-15
DC 5 53 DX
DE 54 5 DX
DLP 90 91 DX
DLN 92 90 DX
DP 4 3 DX
EGND 99 0 POLY(2) 3 0 4 0 0 .5 .5
FB 7 99 POLY(5) VB VC VE VLP VLN 0 53.05E6
```

FIGURE 6-2

628

630

632

FIGURE 6-3

```
+ -50E6 50E6 50E6 -50E6
GA      6  0 11 12 37.70E-6
GCM     0  6 10 99 11.92E-9
IEE     3 10 DC 2.476E-6
HLIM    90  0 VLIM 1K
Q1      11  2 13 QX
Q2      12  1 14 QX
R2       6  9 100.0E3
RC1     4 11 26.53E3
RC2     4 12 26.53E3
RE1     13 10 4.820E3
RE2     14 10 4.820E3
REE     10 99 80.78E6
RO1      8  5 50
RO2      7 99 50
RP       3  4 34.71E3
VB       9  0 DC 0
VC       3 53 DC 2
VE      54  4 DC 5.000E-3
VLIM     7  8 DC 0
VLP     91  0 DC 40
VLN      0 92 DC 40
.MODEL  DX D(IS=800.0E-18)
.MODEL  QX.PNP(IS=800.0E-18 BF=31.58)
.ENDS
V1 1 0 AC=1 PULSE 0 -1 1MS
V2 5 0 DC=-5
V3 6 0 DC=5
.END
```

## \*\*\*\*\*SENSITIVITY DATA\*\*\*\*\*

PARAMETER NAME: r5  
 NOMINAL VALUE: 500.00K

FIGURE 7

VECTOR	SENSITIVITY%
mean(v(4))	1.5111

## \*\*\*\*\*SENSITIVITY DATA\*\*\*\*\*

PARAMETER NAME: r2  
 NOMINAL VALUE: 1.0000Meg

VECTOR	SENSITIVITY%
mean(v(4))	17.265M

## \*\*\*\*\*SENSITIVITY DATA\*\*\*\*\*

PARAMETER NAME: c4  
 NOMINAL VALUE: 10.0000N

VECTOR	SENSITIVITY%
mean(v(4))	-752.77M

## \*\*\*\*\*SENSITIVITY DATA\*\*\*\*\*

PARAMETER NAME: r1  
 NOMINAL VALUE: 10.0000K

VECTOR	SENSITIVITY%
mean(v(4))	-571.46M

## \*\*\*\*\*SENSITIVITY DATA\*\*\*\*\*

PARAMETER NAME: c3  
 NOMINAL VALUE: 10.0000N

VECTOR	SENSITIVITY%
mean(v(4))	4.5201

##### sensitivity analysis results #####  
 tran analysis measurements

test 1 mean  
 mean(v(4)) = 2.086052e-001

Total run time: 0.583 seconds.

Total run time: 0.583 seconds.

Memory remaining = 1996210 Kbytes  
 Memory Used = 14401 Kbytes

## RSS, root summed square analysis Simulation Template With Comments:

*\*Instruct the netlist builder to show tolerances* } 802  
#tolerance

*\*Suppress automatic vector saves* } 804  
#nosave

*\*Suppress lsSpice4 printout* } 806  
#noprint

*\*Save vectors needed for measurements* } 808  
#vector

*\*Set the output file pointer to the beginning to remove* } 810  
*\* the input net list*  
set rewind

*\*Set the noecho environment for print formatting* } 812  
set noecho

*\*Run the specified simulation and save the results* } 814  
#simulation  
set printmode = save  
#mprint

*\*Set the print format* } 818  
SET COLWIDTH=22  
SET SPICEDIGITS=5

*\*Rename the simulation plot* } 816  
nameplot ref

*\*Loop through all of the parameters*  
nextparam null

*\*Tell the user where we are*  
printstatus -t "##### sensitivity for each parameter #####" } 820  
while param <> null

*\*Alter each parameter*  
alterparam tolerance(param)/3

*\*Simulate*  
#simulation

*\*Save the parameter reference in the new plot*  
paramvec = param

*\*Tell the user where we are*  
printstatus -p paramvec

*\*Save the data*  
#mprint

*\*Loop through the vectors*  
nv = nextvector(null)  
while nv <> null

FIGURE 8-1

800

822

FIGURE 8-2

```

*Save the sensitivities of scalar data
  if length(nv) = 1
    nv = nv - ref.nv
  end ; end if
*Get the next vector
  nv = nextvector(nv)
end ; end vector loop
*Restore the parameter
unalterparam
*Get the next parameter
nextparam
end ; end parameter loop
*Set print mode for printing output data
set printmode = print
unset noecho
*Make a new plot for rss data and make it the current plot
newplot rss ref.default ref.default
setplot rss
*Loop through the vectors
nv = nextvector(null)
while nv <> null
  *Initialize the scalar data to zero
  IF LENGTH(NV) = 1
    nv = 0
  END ; end if
  *Get the next vector
  nv = nextvector(nv)
end ; end vector loop
*Loop through the plots
pl = nextplot(null)
*Tell the user where we are
printstatus -t "##### computing rss for each measurement #####"
while pl <> null
  *Select plots
  if sameplot(rss.default) = 0
    if sameplot(ref.default) = 0
      *Tell the user what we are doing
      printstatus -l pl
      *Make the saved parameter reference, paramvec current
      SETPARAM PARAMVEC
      *Get the next vector
      nv = nextvector(null)
      *Print formatted data
      ECHO
      ECHO -U "*****RSS DATA*****"
      ECHO
      ECHO -UN " PARAMETER NAME: "
      PRINTNAME PARAMVEC
      ECHO
      ECHO -UN " NOMINAL VALUE: "
      PRINTVAL PARAMVEC
      ECHO
      ECHO -UN "TOLERANCE VALUE: "
      PRINTTOL PARAMVEC
    end ; end if
  end ; end if
end ; end while

```

824

832

826

828

FIGURE 8-3

```

ECHO
ECHO
PRINTTEXT -UN VECTOR
PRINTTEXT -U SENSITIVITY% RSS_CONTRIBUTION
ECHO
ECHO
*Loop through vectors
while nv <> null
    if length(nv) = 1
        IF REF.NV <> 0
            *Calculate the RSS percentage if value is not zero
            PRINTNAME NV
            NEWNV = (300*NV) / REF.NV
            PRINTVAL NEWNV
        ELSE
            *Calculate the RSS if value is not zero
            PRINTNAME NV
            NEWNV = 3*NV
            PRINTVAL NEWNV
            ECHO -N *
        END ;end if
        *Save and print the sum of squares
        rss.nv = rss.nv + nv * nv
        PRINTVAL RSS.NV
        ECHO
    end ;end if
    * Get next vector
    nv = nextvector(nv)
end ;end vector loop
* Sort plot by descending value
sort -vd
end ;end if
end ;end if
* Get next plot
pl = nextplot(pl)
end ;end plot loop
*Sort the rss plot by descending value
setplot rss
SORT -VD
*Print Headers
ECHO
ECHO -U "*****RSS HI/LO ANALYSIS RESULTS*****"
ECHO
PRINTTEXT -UN VECTOR
SET COLWIDTH=15
PRINTTEXT -U NOMINAL RSS-VALUE TOLERANCE% HI_VALUE LO_VALUE
ECHO
ECHO
*Make a new plot for results
newplot hirss ref.default ref.default
*Loop through the vectors
nv = nextvector(null)
while nv <> null
    if length(nv) = 1

```

834



FIGURE 8-4

```

*Print formatted data
SET COLWIDTH=22
    PRINTNAME NV
SET COLWIDTH=15
PRINTVAL REF.NV
    NV = 3 * SQRT (ABS (NV) )
PRINTVAL NV
    IF REF.NV <> 0
        NEWNV1 = (100*NV) / REF.NV
    ELSE
        NEWNV1 = NV*0
    END
PRINTVAL NEWNV1
    NV = REF.NV + NV
HI_RSS = REF.NV + NV
LO_RSS = REF.NV - NV
PRINTVAL HI_RSS
PRINTVAL LO_RSS
ECHO
end ; end if
*Get next vector
nv = nextvector (nv)
end; end vector loop
ECHO
ECHO
*Print data in output file for SpiceNet to read
setplot hirss
echo ##### RSS HI analysis Results #####
#mprint
RUSAGE ELAPSED

```

830'

**EVA, Extreme Value Analysis Simulation Template With Comments:**

FIGURE 9-1

*\*Instruct the netlist builder to show tolerances* } 902  
#tolerance

*\*Suppress automatic vector saves* } 904  
#nosave

*\*Suppress IsSpice4 printout* } 906  
#noprint

*\*Save vectors needed for measurements* } 908  
#vector

*Set the noecho environment for print formatting*

set rewind - 910

set noecho - 912

*\*Run the specified simulation and save the results* } 914'  
#simulation  
pltype = 0 ;Identify the plot type for later use  
set printmode = save  
#mprint

*\*Set the print format* } 918  
SET COLWIDTH=22  
SET SPICEDIGITS=5

nameplot ref 916  
newplot evahi ref.default ref.default } 932  
evahi.pltype = 0 ;Identify the plot type for later use

*\*Print status for the user*  
printstatus -t "##### sensitivity for each parameter #####" } 920

*\*Loop through the parameters*

nextparam null

while param <> null

*\*Alter each parameter*

alterparam tolerance(param) /3

*\*Simulate, making a new plot for results*

#simulation

*\*Save the current parameter reference*

paramvec = param

*\*Tell the user where we are*

printstatus -p paramvec

*\*Save the tol and paramval*

paramtol = tolerance(param)

paramval = getparam(param)

pltype = 1 ;Identify the plot type for later use

*\*Save the simulation results*

FIGURE 9-2

```

#mprint
*Loop through all the vectors
nv = nextvector(null)
while nv <> null
    *Save the sensitivities for all scalar measurements except pltype
    if length(nv) = 1
        if nv <> pltype
            nv = nv - ref.nv
        end ;end if
    end ;end if
    nv = nextvector(nv)
end ;end vector loop
*Restore the param
unalterparam
*Get the next param
nextparam
end;end parameter loop
*Make ref the current plot
setplot ref
*Loop through the vectors in ref
nv = nextvector(null)
*Tell the user where we are
printstatus -t "##### measurements #####"
while nv <> null
    if length(nv) = 1
        *Loop through all the plots containing scalar vectors
        pl = nextplot(null)
        while pl <> null
            if pltype = 1
                * the inner loop, we are looping through each sensitivity plot looking at the same vector
                * we will alter the parameter id'd by paramvec to maximize/minimize the vector
                setparam paramvec
                *Change each parameter to its worst case extreme value
                if nv >= 0
                    alterparam paramtol
                else
                    alterparam -paramtol
                end
            end
            pl = nextplot(pl)
        end
        *Simulate for the extreme case and save the data in a new plot
        #simulation
        #mprint
        pltype = 2 ;Identify the plot type for later use
        * if we want sensitivity at the extreme, we need to go through each param
        * and change it to be a bit different than it is at the extreme, run a simulation ,
        * and mark it as pltype 3 along with its paramvec, then we can take the diff
        * from the pltype = 2 to get the sensitivity at the extreme if the sign at the extreme
    end
end

```

934

FIGURE 9-3

*\* is different than at the nominal, we can report an error or go on to do worst case*  
*\* for worst case, we need to reduce the param change by 1/2 and do this over again...*  
*\* either continue in this loop or make a wc loop afterward... save the paramvalue*  
*\* and tolerance*

```

    evahi.nv = nv
    if nv <> pltype
        *Tell the user where we are
        printstatus -n nv
    end ;endif

```

```

end ;end plot loop
;get the next vector

```

```

nv = nextvector(nv)

```

```

end ;end vector loop

```

*\*Set print mode and print header*

```

set printmode = print

```

```

unset noecho

```

```

setplot EVAHI

```

```

ECHO

```

```

ECHO -U "*****EVA PARAMETER LIST*****"

```

```

ECHO

```

```

PRINTTEXT -UN PARAMETER

```

```

PRINTTEXT -U NOMINAL TOLERANCE

```

```

ECHO

```

```

ECHO

```

*\*Loop through the parameters*

```

nextparam null

```

```

while param <> null

```

*\*Extract the saved param reference and print its data*

```

    paramvec = param

```

```

    PRINTNAME PARAMVEC

```

```

    PRINTVAL PARAMVEC

```

```

    PRINTTOL PARAMVEC

```

```

    ECHO

```

*\*get the next param*

```

    nextparam

```

```

end ;end param loop

```

```

ECHO ;print a blank line

```

*\*Make a new plot to hold sorted results*

```

newplot evasort ref.default ref.default

```

*\*Make ref the current plot*

```

setplot REF

```

*\*Loop through all vectors in ref*

```

nv = nextvector(null)

```

```

while nv <> null

```

```

    if length(nv) = 1

```

*\*save the result in evasort as a percent of its value*

```

        if ref.nv <> 0

```

```

            evasort.nv = ((evahi.nv-ref.nv)*100)/ref.nv

```

FIGURE 9-4

```

        else
            evasort.nv = 0;
        end ;end if
    end ;end if
    *Get the next vector
    nv = nextvector(nv)
end ;end vector loop
*Print some headers
ECHO
ECHO -U "*****EVA-HI RESULTS*****"
ECHO
PRINTTEXT -UN VECTOR
PRINTTEXT -U NOMINAL EVA-HI CHANGE%
ECHO
ECHO
setplot evasort
*Sort evasort by descanting data
sort -VD
*Loop through the vectors
nv = nextvector(null)
while nv <> null
    *If its the correct data in the correct plot, print it
    if length(nv) = 1
        if nv <> pltype
            PRINTNAME NV
            PRINTVAL REF.NV
            PRINTVAL EVAHI.NV
            PRINTVAL EVASORT.NV
            ECHO
        end ; end if
    end ; end if
    *Get the next vector
    nv = nextvector(nv)
end ; end vector loop
ECHO
ECHO

* now the eva results are in pltype = 2 plots
*Print the results so SpiceNet can read the eva-hi data
set printmode = print
unset noecho
setplot evahi
echo ##### EVA HI analysis Results #####
#mprint

```

940

930

## WCS, Worst Case by Sensitivity Simulation Template With Comments:

*\*Instruct the netlist builder to show tolerances*  
#tolerance } 1002

*\*Suppress automatic vector saves*  
#nosave } 1004

*\*Suppress IsSpice4 printout*  
#noprint } 1006

*\*Save vectors needed for measurements*  
#vector } 1008

*\*Set the output file pointer to the beginning to remove*  
*\* the input net list*  
set rewind } 1010

*\*Set the noecho environment for print formatting*  
set noecho } 1012

*\*Run the specified simulation and save the results*  
#simulation  
set printmode = save  
#mprint } 1014

*\*Set the print format*  
SET COLWIDTH=22  
SET SPICEDIGITS=5 } 1018

*\*Rename the simulation plot*  
nameplot ref } 1016

*\*Make a newplot for results*  
newplot result ref.default ref.default } 1032

*\*Set the current plot to ref*  
setplot ref } 1034

*\*Print status for the user*  
printstatus -t "##### sensitivity for each parameter ##### } 1020

*\*Loop through the parameters*  
nextparam null  
while param <> null  
    *\*Alter each parameter*  
    alterparam tolerance(param)/3  
    *\*Simulate, making a new plot for results*  
    #simulation } 1022'  
    *\*Save the current parameter reference*  
    paramvec = param  
    *\*Inform the user about what's being done*  
    printstatus -p paramvec  
    *\*Make and save the measurements*  
    #mprint

FIGURE 10-1

1000

FIGURE 10-2

```

*Loop through the vectors
nv = nextvector(null)
  while nv <> null
    *Save the sensitivity of scalar quantities
    if length(nv) = 1
      nv = nv - ref.nv
      *Save the worst case -hi value
      result.nv = result.nv + abs(3*nv)
    end ;end if
    nv = nextvector(nv)
  end ;end vector loop
*restore the parameter value
unalterparam
*get the next parameter
nextparam
end ;end parameter loop

*Set the print mode to print instead of save
set printmode = print
*Restore the echo mode for printing
unset noecho
*Set result to the current plot
setplot result
*Print the header
ECHO
ECHO -U "*****WCS PARAMETER LIST*****"
ECHO
PRINTTEXT -UN PARAMETER
PRINTTEXT -U NOMINAL TOLERANCE
ECHO
ECHO
*Loop through the parameters
nextparam null
while param <> null
  paramvec = param
  *Print the row
  PRINTNAME PARAMVEC
  PRINTVAL PARAMVEC
  PRINTTOL PARAMVEC
  ECHO
  nextparam
end
ECHO

*Make a new plot to hold sorted results
newplot wsort ref.default ref.default
*Set the current plot to ref
setplot REF
*Loop through its vectors
nv = nextvector(null)
while nv <> null
  *Calculate the wc as a percent change results
  if length(nv) = 1
    if ref.nv <> 0
      wsort.nv = ((result.nv-ref.nv)*100)/ref.nv
    
```

1036

1038

FIGURE 10-3

```

        else
            wcsort.nv = 0;
        end ;end if
    end ;end if
    nv = nextvector(nv)
end ;end vector loop

```

*\*Print headers*

```

ECHO
ECHO -U "*****WCS-HI RESULTS*****"
ECHO
PRINTTEXT -UN VECTOR
PRINTTEXT -U NOMINAL WCS-HI CHANGE%
ECHO
ECHO

```

```

;sort wcsort by descending value
setplot wcsort
sort -VD

```

*\*Print the ordered list*

```

nv = nextvector(null)
while nv <> null
    if length(nv) = 1
        PRINTNAME NV
        PRINTVAL REF.NV
        PRINTVAL RESULT.NV
        PRINTVAL WCSORT.NV
        ECHO
    end
    nv = nextvector(nv)
end

```

```

end
ECHO
ECHO

```

*\*Set the current plot to the wc results*  
 setplot result

```

echo ##### WCS HI analysis Results #####

```

*\*Print the measured results in a form that can be read back*

*\*into SpiceNet*

```

#mprint

```

*\*Report the elapsed time in the output file*  
 rusage elapsed

1040

1030